REMARKS

35 U.S.C. 112 Rejection

Claim 1 has been amended to provide antecedent basis for the term "said session."

35 U.S.C. 103(a) Rejection

The Examiner has rejected claims 1, 3-5, and 11 under 35 U.S.C. 103(a) as being unpatentable over Schneck et al., (6,260,039) in view of Cavanaugh (5,918,214) and claims 6-9, 13-19 under 35 U.S.C. 103(a) as being unpatentable over Schneck et al. in view of Cavanaugh and further in view of Call (6,154,738).

In reply, Applicant submits that in order to show prima facie obviousness, it is required "that (1) there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings; (2) there must be a reasonable expectation of success; and (3) the prior art references teach or suggest all the claim limitations" (MPEP 2143).

Claims 1, 3-5, 11

The examiner proposed that it would have been obvious to one ordinary skilled in the art at the time the invention was made to implement the claimed invention because Schneck and Cavanaugh teach substantial features of the claimed invention. Applicant respectfully disagrees.

With respect to the amended claim 1, Cavanaugh fails to teach, describe, or suggest the following limitations:

(a) "a directory of identifiers and metadata to a plurality of network services, wherein said network services receive XML inputs and produce XML outputs;"

(b) "an engine for receiving requests, using said identifiers in said directory to direct said requests to access said network services when requested, and constructing a state storing session for interfacing with said network services by using one of said network services to transform an XML runtime model into said state storing session, wherein said runtime model defines the interaction between said state storing session and said network services, wherein said state storing session uses a driver to interface with each of said network services via stateless network protocols and said state storing session is configured from said metadata from said directory; and"

(c) "a plurality of service providers accessible to said plurality of drivers for providing network services identified in said directory."

In regard to limitation (a), the Examiner stated that Cavanaugh teaches a directory (52) of identifiers (152, 160) and metadata (158-164). In Cavanaugh, naming service 52 contains object reference 150 (col. 7, lines 25-29, col. 10, lines 17-18). The information found in object reference 150 contains information about objects, which are accessed by client requests (FIG. 5, col. 10, lines 16-35). However, in the present invention the identifiers and metadata in the directory are related to network services. Thus the only way that Cavanaugh can anticipate this limitation of the present invention is if objects in Cavanaugh are equivalent to network services of the present invention. However, the Examiner had indicated that features, not objects, in Cavanaugh are network services (col. 1, line 65 - col. 2, lines 8). Furthermore, the alleged network services identified by the Examiner in Cavanaugh (col. 1 line 65 to col. 2 line 8) are system features that support the execution of the objects within Cavanaugh. Fig. 4 of Cavanaugh shows that these features are grouped into subcontracts (also see col. 2, line 26), which are distinct from objects (e.g. printer and modem object implementations). As shown in the two registries 250 and 200 in Figs. 2 and 3 (col. 14, lines 4-26), pointers to subcontracts are made by

developers in creating object implementations. Col. 2 lines 34-35 state that "subcontracts are separate from object interfaces and object implementations." Furthermore, compared to the amended claim 1, Cavanaugh does not teach "network services" that "receive XML inputs" and "produce XML outputs." Thus, Cavanaugh does not teach the limitation "directory of identifiers and metadata to a plurality of network services."

In regard to limitations (a) and (b), it is noted that neither features nor objects disclosed in Cavanaugh are equivalent to network services for the following reasons. First, in practice, in Cavanaugh there is no way to access these features without instantiating objects via the COBRA protocol. Also, the features disappear after the objects are deactivated. In contrast, the network services in the present invention are not instantiated objects and require no layer of COBRA or any such object management protocol. Second, the present invention can take advantage of existing loosely coupled network services via standard stateless internet protocols such as HTTP. Such a protocol sends inputs and outputs across the network and maintains no state information in between transactions. For example, the server objects in Cavanaugh must be maintained as long as the stub functions and other supporting objects are maintained in the client side. In contrast, the network services in the present invention can send back replies and be freed of any further obligation to maintain state or allocate object resources, even if the state-storing session has not been de-allocated. In other words, the network services are not tied to the state-storing session and hence do not have to maintain and synchronize state information with it outside of the atomic transactions defined in the internet protocols (e.g. requests and replies). This is reflected in the amendments to limitation (b), which now includes "via stateless network protocols."

In regard to limitation (b), Cavanaugh does not teach the limitation of "using a network service to transform an XML runtime model" in the construction of a state storing session. In the

present invention, a XML runtime model is transformed to create the state storing session (referred to as a "Web Services Application" in the specification). The XML runtime model defines the behavior of the state storing session. Cavanaugh uses no such XML runtime model in response to the incoming requests. Its subcontract layer 36 does not have such a mechanism. It is noted that although the Call reference contains disclosure to XML, it does not teach the use of a runtime model written in XML used for the construction of a state storing session as disclosed in the present invention.

Furthermore, the alleged directory, naming service (52), is not used by Cavanaugh to provide for the configuration of the alleged state storing session, subcontract layer 36. Cavanaugh makes no mention that subcontract layer 36 is configured. The distinction must be made between subcontracts, which are implemented by the programmer ahead of time and hence manually configurable, and subcontract layer 36, which "provides the functionality required by an object in order to utilize subcontracts to implement various services (or features or object mechanisms) named by a particular subcontract" (col. 9, lines 37-39).

The examiner has also proposed that requests are received in Cavanaugh for the invocation of objects. In particular, object adapter 28 (col. 5, lines 29-35) is mentioned as an interface to network services. Yet, as mentioned before, the Examiner has stated that features, not objects, are network services. Since object adaptor 28 does not interface with features, Cavanaugh does not teach "an interface to network services." Instead, Cavanaugh states that the object adapter 28 interface with ORB objects 14. The aforementioned argument with regard to the differences between the objects/features in Cavanaugh again applies here. For these reasons, Cavanaugh does not teach every limitation disclosed in claim 1.

As stated in Applicant's last reply to office action, Schneck does not teach a method/system comprising storing identifiers and metadata of a plurality of services in a

directory. Specifically, with regard to "identifiers in a directory", the Examiner cited that Schneck teaches "storing identifiers of a plurality of network services (col. 4/lines 23-33, 43-63, col. 3/lines 37-38) in a (104) directory (col. 3/lines 37-38)." And with regard to "metadata", the Examiner cited that Schneck teaches "metadata descriptive information, i.e. metadata" in a previous rejection of claim 2, citing col. 3/lines 57-60, which states:

> ... 202, which responds to the requests; map 212, which correlates the requests to the template files (i.e. request mapping) and correlates abbreviated names to unabbreviated names (i.e. friendly name mapping); and template files 214...

However, Applicant submits that in Schneck's teaching, map 212 is not the same as directory 104 (Fig. 2). As such, even if the cited reference of map 212 contains metadata, it is not in a directory (104) with identifiers, as cited by the Examiner. They are separate entities and Schneck does not contain any teaching to combine them for single location storage, as is the case with the directory of the present invention. Furthermore, Fig. 3 of Schneck clearly teaches that accesses to directory (104) and map (212) occur in two different steps (302 and 304) (col. 5 lines 1-7). In contrast, the present invention stores identifiers and metadata in a single directory and the access is made in one step.

As stated in limitation (b), the engine uses the metadata for several operations when a request is accepted. In Schneck, there is no mention of session tracking, wherein multiple requests can access a state storing session as in the present invention. Furthermore, each session is configured by using the metadata stored in the directory. Schneck does not teach any of such use of metadata.

Finally, as the Examiner stated, Schneck "does not explicitly teach using said identifiers to access said network services." Applicant maintains that an important distinction needs to be made here about the content of the directory. While both the present invention and Schneck's invention have data in their respective directories, the data stored in Schneck's directory are

contact information and organizational information and do not identify any network service external to the directory. All incoming requests receive responses composed of information from Schneck's directory. In contrast, the data in the present invention's directory are identifiers used for accessing network services.

Applicant thus contends that Cavanaugh and Schneck, either alone or in combination, do not teach the every element claimed invention. Furthermore there is no suggestion in the references to modify them to make the claimed invention. Applicant asserts that the 103(a) rejection on claim 1 has been overcome.

As claims 3-5 depend from claim 1 or another claim that depends on claim 1, these claims are in a condition for allowance as well. Their rejection based upon 35 U.S.C. 103(a) has been overcome.

Claims 6-10, 12-20

Claim 12 has been cancelled. As for the rest of the claims, the examiner proposed that it would have been obvious to combine Schneck, Cavanaugh, and Call to make the claimed invention. With respect to claim 6-10, Call does not teach the limitations of claim 1 that both Schneck and Cavanaugh fail to teach. Call does not teach or suggest any of the limitations from claim 1. Specifically, there is no mention of a directory in Call. Thus, even in combination with Call, the prior art does not teach all the limitations of claim 1. As claims 6-10 all depend on claim 1, Applicant asserts that their rejection based on 103(a) is overcome.

Claim 13-20 overcomes the 103(a) rejection for the same reason as claim 6-10, as claims 13-20 depend on claim 11. Applicant submits that they are in a condition of allowance as well.

CONCLUSION

The Examiner has rejected claims 1, 3-9, 11, and 13-19 under 35 U.S.C. 103(a). In response, Applicant has amended claims 1 and 11 and responded to the 35 U.S.C. 103(a) and 35 U.S.C. 112 rejections on pending claims 1, 3-9, 11, and 13-19. Applicant asserts that the present application is in a condition for allowance.

Respectfully submitted,

COUDERT BROTHERS LLP

By: _____
David Chan
Reg. No. 51,540

COUDERT BROTHERS LLP
333 South Hope Street
Suite 2300
Los Angeles, California 90071
(213) 229-2990